Hypermedia Systems Development:
Do We Really Need New Methods?

Michael Lang
National University of Ireland, Galway

Michael.Lang@nuigalway.ie

Abstract

Hypermedia systems design presents challenges that are not normally encountered with the development of orthodox 'traditional' information systems. In recognition of these challenges, and of the purported inadequacy of conventional software engineering techniques, there is much support in the literature for the view that new, specialized methods are needed. The question must therefore be asked: do we really need new methods for hypermedia systems design? This research-in-progress paper provides an outline answer by asking how hypermedia systems differ significantly from 'traditional' information systems, by hypothesizing why specialized hypermedia design methods set forth in the literature are not being use in practice, and by suggesting a number of implications.

Keywords: Hypermedia; Systems Development Methods; Design Techniques

Introduction

Hypermedia systems design presents challenges that are not normally encountered with the development of orthodox 'traditional' information systems. In recognition of these challenges, and of the purported inadequacy of conventional software engineering techniques, there is much support in the literature for the view that new, specialized methods are needed.

However, just as Nanard & Nanard (1995) set forth the claim that fundamental differences between hypertext design and software engineering "make a pure transposition of techniques both difficult and inadequate", there is the counter-argument by Keen (1991) that "in practice, very few of the major areas of concern in ISR are less than twenty years old …. issues seen as 'new' turn out to have long roots".

Much of the literature on Web and hypermedia development fails to appreciate the legacy of traditional IS development and other root disciplines such as, for example, relational and network database design, visual event-driven programming, graphic design, industrial design, and media production. Furthermore, experiences in traditional information systems (IS) development reveal that formalized methods have been largely rejected, and even in cases where formalized methods are used, there is typically some degree of adaptation (Fitzgerald, 1997; Hardy et al., 1995).

The question must therefore be asked: do we really need new methods for hypermedia systems design? This research-in-progress paper provides an outline answer by asking how hypermedia systems differ significantly from 'traditional' information systems, by hypothesizing why specialized hypermedia design methods set forth in the literature are not being use in practice, and by suggesting a number of implications.

What is so Different about Hypermedia Systems?

Firstly, the issue must be raised: what is so different about hypermedia systems? Some of the points in the following discourse relate just to Web-based hypermedia systems, as opposed to hypermedia systems in general, but as the Web is the most common platform for hypermedia systems they are included.

Complexity of Hypermedia Structures and Multimedia Data

Hypermedia systems are inherently much more complex than conventional systems. Essentially, hypermedia attempts to emulate the intricate mechanisms of the human mind by associating blocks of knowledge with each other in a complex multitude of "associative trails" (Bush, 1945). Not surprisingly, there is a tendency towards chaotic 'spaghetti' structures, rather like the much-maligned practice of *goto* programming. Arbitrary linking, like the arbitrary use of *goto*, "is just too primitive; it is too much of an invitation to make a mess" (Dijkstra, 1968; de Young, 1990). Even within small systems this quickly becomes a problem. For large-scale organisational systems, such issues as 'getting lost in cyberspace', locating information, visualising knowledge structures, and managing content are major considerations.

Moreover, there is the added complexity of storing, processing and presenting multimedia data. No longer do information systems merely handle simple text and numeric data types, but must now cater for graphics, images, audio and video objects, and compound media. These more sophisticated data types, particularly those that have temporal aspects, present further challenges to systems programmers, database administrators and interface designers.

Interdisciplinary Collaboration

Hypermedia systems development is characterized by a lack of consensus because of its diverse roots. Development teams typically involve people from a broad range of backgrounds, often with little in common. Of course, skills diversity is not unique to hypermedia systems development, - many conventional projects, particularly large ones, necessitate the integration of various knowledge domains. However, participants in hypermedia systems development tend to hail from disciplines that are not just diverse but also discrete, – such as systems analysis, graphic design, marketing, and media production. These disciplines each have their own very different vocational training paths, with little crossover. Inevitably, this leads to communication problems and cultural conflicts. This is especially common between software engineers and graphic designers, two communities that appear to operate in wholly different worlds (Gallagher & Webb, 1997).

"Web-Time" Product Development Cycles

It has been reported (Fitzgerald, 1997) that typical IS development projects now comprise about three developers for less than six months, which seems to reflect a profile of small-scale, rapid development. However, the advent of the Web has accelerated this even further. By its very nature, the Web is an immediate medium. Web developers are not impeded by production and distribution delays such as characterize the implementation of shrink-wrapped software. Web-based systems can be easily and quickly launched by developing functional front-end interfaces, powered by crude but effective back-end software, which can later be modified and enhanced in such a manner that end users may be oblivious to the whole process. For projects operating in "Web time", time-to-market is typically between two and three months (Barry & Lang, 2001b). Thomas thus characterizes Web-based application development as "guerilla programming in a hostile environment using unproven tools, processes, and technology" (Thomas, 1998).

External Focus of Web-based Information Systems

Traditionally, information systems served internal functions within organisations. With the advent of the Web, organisations are being turned inside-out as Web-based systems necessarily have an external focus. In essence, Web-based information systems have become the interface between organisations and the environments within which they operate, - shop windows to the world, as it were. Poorly designed systems stand to be exposed in all their shabbiness to a global user base. The usability of Web-based information systems therefore becomes a critical issue (Buckingham Shum & McKnight, 1997), and programmers and interface designers can no longer afford to pay indignant disregard to usability factors. Furthermore, unlike traditional information systems, there is no definite group of known users who are waiting upon the system and from whom the requirements emanate. Therefore, marketing personnel must be involved to a greater extent, both in discovering user requirements and in promoting the site once implemented.

Ongoing Maintenance

Because the content and structure of hypermedia systems is inherently dynamic, maintainability is crucial and involves significant overheads. Isakowitz (1993) speaks of the problems of "pollution" and "waste" that can arise with the growth of hypermedia systems. Hypermedia systems, particularly those that are Web-based, undergo much more continuous, fine-grained evolution than traditional systems (Cusumano & Yoffie, 1998). It is therefore important that in developing hypermedia systems, due consideration is taken of system growth, so that it can occur in a controlled, but flexible and consistent manner.

What is Wrong with Hypermedia Design Methods?

On the basis of the points previously outlined, it may be argued that because hypermedia systems (as a general classification) are in many ways different from conventional information systems, different methods are necessitated. Quite a number of specialized methods for hypermedia systems development have been proposed in the literature, the best known of which are presented in Table 1 and Figure 1.

However, a recent study has found that none of these methods are being used to any significant extent by practitioners (Barry & Lang, 2001b). It is unclear why this is so, but the following suggestions are tentatively set forth as possible explanations.

Lack of Awareness

Perhaps the most obvious explanation is that practitioners are unaware that these methods exist. It is well acknowledged that practitioners have little interest in academic journals or conferences, and that it is

Relationship Management Methodology (RMM)	Isakowitz et al. (1995)
Object-Oriented Hypermedia Design Methodology (OOHDM)	Schwabe & Rossi (1995)
Enhanced Object-Relationship Model (EORM)	Lange (1994)
World Wide Web Design Technique (W3DT / SHDT)	Bichler & Nusser (1996a)
Extended World Wide Web Design Technique (eW3DT)	Scharl (1999)
Web Site Design Method (WSDM)	De Troyer & Leune (1997)
Scenario-based Object-Oriented Hypermedia Design Methodology (SODHM)	Lee et al. (1999a)
View-Based Hypermedia Design Methodology (VHDM)	Lee et al. (1999b)

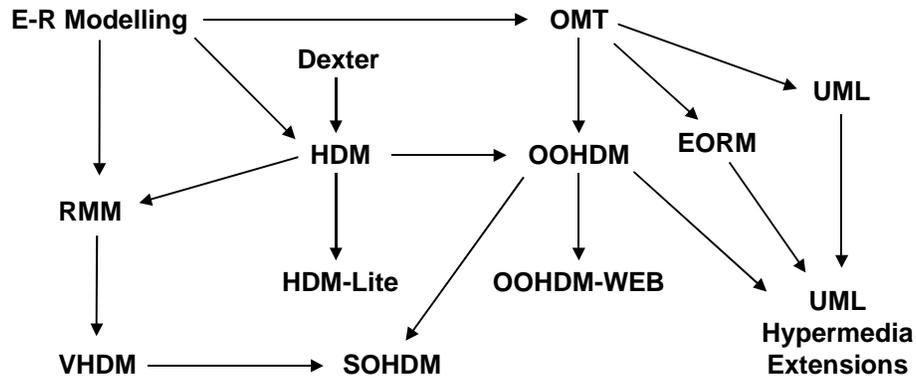**Table 1: Hypermedia Development Methods.**

Figure 1: Evolution of Hypermedia Development Methods.

books that move a discipline onwards (Keen, 1991). Yet, all published references to these methodologies appear either in academic journals or conference proceedings. Furthermore, it is only in recent years that specialized third-level programmes in Web and multimedia development have been established by universities, so it will be some time before the results of teaching filter into practice. Of course, many of the aforementioned methods are unknown not just to practitioners but also to academics, in which case they do not form part of the curricula.

Inertia

The software development industry is notoriously slow to change. Barry & Lang (2001a; 2001b) report much ongoing usage of the classical Systems Development Life Cycle (SDLC) and variants, and of legacy techniques such as data flow diagramming, for multimedia systems development. Conversely, their study reveals little usage of newer methods and techniques such as RMM, OOHDM and the Unified Modelling Language (UML). It therefore seems that inertia may be an explanatory factor, as developers are reluctant to abandon older methods and techniques even when their usefulness is questionable.

As it stands, there is already a multitude of software modelling techniques in use in general practice (Gray & Rao, 1993). For far too long, the reviewers of academic publications have willingly accepted papers outlining yet more "new" methods, so that now the literature is littered with a plethora of methods and techniques, many of which are quite similar. In this regard, the development of the UML standard to integrate the various conflicting and confusing object-oriented notations is to be greatly welcomed, and it is hoped that this example in striving for unification rather than fragmentation shall be followed by future method developers.

In the context of hypermedia design, rather than adding to the confusion by creating altogether new techniques, it is better to assess which of the techniques from traditional systems development are appropriate as is, which may be adapted, and which should be replaced entirely. Although many of the techniques in popular usage today hark back to the 1970's (Barry & Lang, 2001b; Fitzgerald, 2000), it would be wrong to assume that merely because those techniques were devised in response to the problems of a different era, they cannot be adapted to the problems of today. In fact, "legacy" techniques such as Entity-Relationship Diagramming (ERDs), Jackson Structured Programming (JSP), and flowcharts may easily be adapted to model aspects of a hypermedia system. It has generally been the case that those parts of techniques that require adaptation are usually small when compared to what remains applicable.

Lack of Tool-based Support

Though Britton et al. (1997) some time ago indicated a real urgency to provide tool-based support for structured modeling notations that are appropriate for multimedia and hypermedia systems, such tools are still rare. What few tools there are that support hypermedia modelling techniques and methods are almost without exception incomplete research prototypes that have not been commercially released into the public domain, such as *RMCase* (Díaz et al., 1995), *WebComposition* (Gellersen & Gaedke, 1999), W3DT *WebDesigner* (Bichler & Nusser, 1996b), *CGI-Lua* (Schwabe & de Almeia Pontes, 1998), and *Araneus* (Atzeni et al., 1998). Furthermore, traditional CASE tools are ill-suited to hypermedia systems development (Arndt, 1999).

Garzotto & Paolini (1993) expressed the aspiration that their HDM approach would be a step towards the development of application generators, but this has not happened in practice. Research models, experimental tools and document standards have thus far failed to have any real impact on existing technology or to trigger the emergence of a new generation of tools. In the absence of appropriate Computer-Aided Software Engineering (CASE) tools, many developers are reliant on manual, paper-based techniques and ad hoc methods (Barry & Lang, 2001b; McClure, 1998).

Insufficient Documentation and Method Complexity

Quite a few of the methods in Table 1 have been applied only rarely outside of academic contexts, or adequately tested in live situations. Because most of them are documented only in academic papers of 3,000 to 5,000 words, there are insufficient illustrated examples of the methods at work. Some, such as OOHDM, are very complex and difficult to grasp, and it is therefore not surprising that in the absence of documentation and user support, those methods are not used.

Barry & Lang (2001b) report that understandability, ease-of-use, and widespread acceptance and reputation amongst developers are major issues in method selection for Web and multimedia systems development. It is significant that most of these methods have been developed by the software engineering community, for use by software engineers. Methods such as RMM and OOHDM which use formalized diagrammatic notations and require specialized training are unlikely to ever be used by graphic designers, marketing personnel, end users, or other stakeholders from backgrounds other than software engineering.

Are Methods Appropriate at All?

Perhaps the most obvious question is: do we need methods at all? The reason that methods are not being used may simply be because their underlying philosophy is severely misguided. It may be argued that formalized hypermedia design methods, founded as they are upon artificial concepts of rationality, are incongruent with their context of use, and that this is why they have not been adopted by practitioners. It is paradoxical to suggest that, on the one hand, hypermedia information systems must operate within highly flexible business environments, and that, on the other hand, the methods to develop such systems must be rigid and systematic.

More than three decades since it was first discussed, the issue of whether or not it is sensible to speak of "software engineering" remains contentious (Jackson, 1998). Software design, - and, in particular, the design of interactive software, - may be said to be a "wicked problem" (Buchanan, 1992). Wicked problems do not deal with true or false answers to questions of fact or logic, and therefore cannot be resolved by rational methods. Rather, designers must rely on informed intuition to resolve difficult dilemmas. This lends support to the viewpoint that information systems development is a creative, problem-solving activity much more akin to a craft than an engineering discipline.

There is a growing body of opinion that systems development is essentially amethodical (Baskerville et al., 1992; Whitley, 1998), and that it is based not on mere compliance with prescribed methods but rather

more on the intuition of experienced developers as to how to act within the unique constraints of individual projects. Ciborra (1999) refers to this as "improvisation", something that is …

> "… *simultaneously rational and unpredictable; planned but emergent; purposeful but opaque; effective but irreflexive; discernible after the fact, but spontaneous in its manifestation*".

This school of thought does not accept that formalized design methods can be executed objectively. Moreover, it holds that no method can ever be better than the people who apply that method, and that, consequentially, methods cannot be applied in a general way; instead, they must always be related to the situation at hand and the people involved (Lowgren & Stolterman, 1999). Accordingly, methods, and techniques should properly be construed as constituting a 'toolkit' in the software designer's arsenal, which he may proactively mix, match and adapt contingent upon a particular problem context (Benyon & Skidmore, 1987).

Implications and Conclusions

Hypermedia systems are inherently complex, require the coordinated involvement of multidisciplinary teams, often need to be very rapidly developed, yet must be robust and usable at the same time. It would therefore reasonably appear that the development of hypermedia systems cannot be performed without at least some overarching approaches, techniques, methods, and tools to guide the process. Although formalized methods seem inappropriate for hypermedia systems development, wholly amethodical approaches cannot be solely relied upon as they require affirmative action for success.

For methods to be of use, they must be framed at a high level of granularity; that is, they should prescribe broad guidelines rather than a complex, over-intellectualized list of tasks (Fitzgerald, 2000). Barry & Lang (2001b) reveal that Web/multimedia systems developers clearly believe that adding structure to the development process is desirable and that they expect to move towards increased adoption of methods, but that they do not want cumbersome or expensive methods. Lean approaches such as rapid prototyping, timeboxing, Extreme Programming (XP), concurrent engineering, and Just-in-Time production seem best suited to the era of Web-time, as they combat complexity, enable flexibility, and reduce exposure to risks by using small dedicated teams, component-based development, layered architectures, and short sharp bursts of incremental work.

If methods and techniques proposed by academics are to make any significant contribution to practice, they must be mapped into usable tools which effectively implement them. Carstensen & Vogelsang (2001) argue that such methods should not be too complex or require specialized skills in formal modeling and specification. The restrictive notations of CASE tools has been seen in the past to be one of the major reasons for non-use. It is therefore encouraging to see developments in CASE technologies which support informal techniques such as storyboarding and interface sketching (Bailey, 1999; Harada et al., 1996; Newman & Landay, 2000).

Conversely, it is not enough that hypermedia design methods and techniques should be supported by tools, but ideally that tools should provide explicit guidance for methods and techniques. For development tools to be truly effective, they must provide support for methods and techniques that aid development teams to conceptualize system architectures at different levels of abstraction. Such tools should support iterative top-down and bottom-up design, and should move beyond the constraints of 2-dimensional modeling techniques by providing 3-dimensional visualization mechanisms.

The underlying premise of method developers seems to be that methods can and do lead to better usability, reusability, cost, quality, requirements closure, maintainability, robustness, and so forth. The question must be asked: where is the evidence to support these claims? Hypermedia design methods are unlikely to be used merely for their own sake, so it shall be necessary to collate empirical data based on experiences in real projects. For this to happen shall require academics to push these methods out into the real world

by publicizing their work in more open forums, by engaging in consultancy, and by conducting qualitative evaluations such as case studies or action research.

References

Arndt, T. (1999) The Evolving Role of Software Engineering in the Production of Multimedia Applications. In *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, Florence, Italy, June 7-11 1999. IEEE.

Atzeni, P., Mecca, G. & Merialdo, P. (1998) Design and Maintenance of Data-Intensive Web Sites. In *Schek, H.-J et al. (Eds.), Proceedings of the Sixth International Conference on Extending Database Technology*, Valencia, Spain, March 1998, pp. 436-450.

Bailey, B. (1999) Interactive Sketching of Multimedia Storyboards In *Proceedings of ACM Conference on Multimedia* Orlando, Florida, October 30 - November 5 1999. ACM Press.

Barry, C. & Lang, M. (2001a) Contrasting Approaches: A Study of Traditional and Multimedia Information Systems Development in Ireland. In *Proceedings of Irish Academy of Management Annual Conference*, Derry, Ireland, 6-7 September 2001.

Barry, C. & Lang, M. (2001b) A Survey of Multimedia and Web Development Techniques and Methodology Usage. *IEEE Multimedia.* April-June 2001**,** 52-60.

Baskerville, R., Travis, J. & Truex, D. (1992) Systems without Method: The Impact of New Technologies on Information Systems Development Projects. In: *Kendall, K. E. et al. (Eds.), The Impact of Computer Supported Technologies on Information Systems Development*, pp. 241-269. Elsevier Science Publishers B.V. (North-Holland).

Benyon, D. & Skidmore, S. (1987) Towards a Tool Kit for the Systems Analyst. *The Computer Journal.* 30(1)**,** 2-7.

Bichler, M. & Nusser, S. (1996a) *Developing Structured WWW-Sites with SHDT*, Vienna University of Economics and Business Administration, Austria.

Bichler, M. & Nusser, S. (1996b) Modular Design of Complex Web-Applications with W3DT. In *Proceedings of the IEEE Fifth Workshop on Enabling Technology: Infrastructure for Collaborative Enterprises (WET ICE '96)*, Stanford, California, USA, June 19-21 1996. IEEE.

Britton, C., Jones, S., Myers, M. & Sharif, M. (1997) A survey of current practice in the development of multimedia systems. *Information and Software Technology.* 39**,** 695-705.

Buchanan, R. (1992) Wicked Problems in Design Thinking. *Design Studies.* 8(2)**,** 5-21.

Buckingham Shum, S. & McKnight, C. (1997) World Wide Web usability: introduction to this special issue *International Journal of Human-Computer Studies.* 47(1)**,** 1-4.

Bush, V. (1945) As We May Think. *The Atlantic Monthly.* July**,** 101-108.

Carstensen, P. H. & Vogelsang, L. (2001) Design of Web-based Information Systems - New Challenges for Systems Development? In *Proceedings 9th European Conference on Information Systems*, Bled, Slovenia, June 27-29 2001, pp. 536-547.

Ciborra, C. U. (1999) A Theory of Information Systems Based on Improvisation. In: *Wendy L. Currie, Bob Galliers (Eds.), Rethinking Management Information Systems*, pp. 136-155. Oxford University Press.

Cusumano, M. A. & Yoffie, D. B. (1998) *Competing on Internet Time / Lessons from Netscape and Its Battle with Microsoft.* The Free Press, New York.

De Troyer, O. M. F. & Leune, C. J. (1997) WSDM: a user centered design method for Web sites. In *Proceedings of Seventh International World Wide Web Conference (WWW7)*, Brisbane, Australia, April 14-18 1998.

de Young, L. (1990) Linking Considered Harmful. In *Proceedings of the European Conference on Hypertext*, INRIA, France, November 1990, pp. 238-249.

Díaz, A., Isakowitz, T., Maiorano, V. & Gilabert, G. (1995) RMCase: A Tool to Design WWW Applications. In *Proceedings of Fourth International World Wide Web Conference*, Boston, MA, December 11-14, 1995.

Dijkstra, E. W. (1968) Go To Statement Considered Harmful. *Communications of the ACM.* 11(3)**,** 147-148.

Fitzgerald, B. (1997) The Use of Systems Development Methodologies in Practice: A Field Study. *Information Systems Journal.* 7(3)**,** 201-212.

Fitzgerald, B. (2000) *Systems Development Methodologies: The Problem of Tenses.* 13(3), 174-185.

Gallagher, S. & Webb, B. (1997) Competing Paradigms in Multimedia Systems Development: Who Shall Be The Aristocracy? In *Proceedings of Fifth European Conference on Information Systems*, Cork, Ireland, June 19-21 1997.

Garzotto, F. & Paolini, P. (1993) HDM - A Model-Based Approach to Hypertext Application Design. *ACM Transactions on Information Systems.* 11(1), 1-26.

Gellersen, H.-W. & Gaedke, M. (1999) Object-Oriented Web Application Development. *IEEE Internet Computing.* Jan-Feb**,** 60-68.

Gray, E. M. & Rao, G. (1993) Software Requirements Analysis and Specification in Europe: An Overview. In: *Thayer, Richard H.; McGettrick, Andrew D. (Eds.), Software Engineering: A European Perspective*, pp. 78-96. IEEE Computer Society Press, Los Alamitos, CA.

Harada, K., Tanaka, E., Ogawa, R. & Hara, Y. (1996) Anecdote: A Multimedia Storyboarding System with Authoring Support. In *Proceedings of ACM Conference on Multimedia* Boston, MA, USA, November 18- 22 1996, pp. 341-351.

Hardy, C. J., Thompson, J. B. & Edwards, H. M. (1995) The use, limitations and customization of structured systems development methods in the United Kingdom. *Information and Software Technology.* 37(9)**,** 467-477.

Isakowitz, T. (1993) Hypermedia, Information Systems and Organizations: A Research Agenda. In *Proceedings of the Twenty-Sixth Annual Hawaii International Conference on System Sciences*, January 1993, pp. 361-369. IEEE Computer Society Press.

Isakowitz, T., Stohr, E. A. & Balasubramanian, P. (1995) RMM: A Methodology for Structured Hypermedia Design. *Communications of the ACM.* 38(8)**,** 34-44.

Jackson, M. (1998) Will There Ever Be Software Engineering ? *IEEE Software.* Jan-Feb**,** 36-39.

Keen, P. G. W. (1991) Relevance and Rigor in Information Systems Research: Improving Quality, Confidence, Cohesion and Impact. In: *Information Systems Research: Contemporary Approaches and Emergent Traditions*, pp. 27-49. Elsevier Science Publishers (North-Holland).

Lange, D. B. (1994) An Object-Oriented Design Method for Hypermedia Information Systems. In *Proceedings of the 27th Annual Hawaii International Conference on System Sciences (HICSS '94)*, pp. 366-375. IEEE Computer Society Press.

Lee, H., Lee, C. & Yoo, C. (1999a) A scenario-based object-oriented hypermedia design methodology. *Information & Management.* 36**,** 121-138.

Lee, H., Kim, J., Kim, Y. G. & Cho, S. H. (1999b) A View-Based Hypermedia Design Methodology. *Journal of Database Management.* 10(2), 3-13.

Lowgren, J. & Stolterman, E. (1999) Design Methodology and Design Practice. *ACM interactions.* Feb**,** 13-20.

McClure, S. (1998) *Web Application Development Developer Perspectives: An IDC White Paper*. International Data Corporation, Framingham, MA.

Nanard, J. & Nanard, M. (1995) Hypertext design environments and the hypertext design process. *Communications of the ACM.* 38(8)**,** 49-56.

Newman, M. W. & Landay, J. A. (2000) Sitemaps, Storyboards, and Specifications: A Sketch of Web Site Design Practice. In *Proceedings of ACM Symposium on Designing Interactive Systems*, Brooklyn, New York, pp. 263-274.

Scharl, A. (1999) A Conceptual, User-Centric Approach to Modeling Web Information Systems. In *Proceedings of Fifth Australian World Wide Web Conference (AusWeb99)*, Ballina, New South Wales, Australia, April 17-20, 1999.

Schwabe, D. & Rossi, G. (1995) The Object-Oriented Hypermedia Design Model. *Communications of the ACM.* 38(8)**,** 45-48.

Schwabe, D. & de Almeia Pontes, R. (1998) *OOHDM-WEB: Rapid Prototyping of Hypermedia Applications in the WWW*. Department of Informatics, Pontifícia Universidade Católica do Rio de Janeiro, MCC 08/98, March 1998.

Thomas, D. (1998) Web Time Software Development. *Software Development Magazine.* October**,** 78-80.

Whitley, E. A. (1998) Method-ism in Practice: Investigating the Relationship between Method and Understanding in Web Page Design. In *Proceedings of Nineteenth Annual International Conference on Information Systems*, Helsinki, Finland, December 13-16 1998.

Biography

Michael Lang has been a Lecturer in Information Systems within the School of Business & Economics at the National University of Ireland, Galway since 1996. Prior to that, he worked in industry as an analyst/programmer. His research interests are business systems analysis and design; hypermedia and multimedia information systems development; requirements management; and IS education. He is a B.Comm. graduate of University College Dublin, and received his M.Sc. in Applied Computing & Information Technology from the National University of Ireland, Galway. He is currently undertaking a Ph.D. in Information Systems at University College Cork.